Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	eZ80
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	24
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/ez80f92az020sc

Introduction . . . 162
ZDI-Supported Protocol . . . 163
ZDI Clock and Data Conventions . . . 164
ZDI Start Condition . . . 164
ZDI Register Addressing . . . 165
ZDI Write Operations . . . 166
ZDI Read Operations . . . 167
Operation of the eZ80F92 Device During ZDI Breakpoints . . . 168
Bus Requests During ZDI DEBUG Mode . . . 169
ZDI Write Only Registers . . . 170
ZDI Read Only Registers . . . 171
ZDI Register Definitions . . . 171

On-Chip Instrumentation . . . 187
Introduction to On-Chip Instrumentation . . . 187
OCI Activation . . . 187
OCI Interface . . . 188
OCI Information Requests . . . 189

Random Access Memory . . . 190
RAM Control Registers . . . 192

Flash Memory . . . 193
Flash Memory Arrangement in eZ80F92 . . . 193
Flash Memory Arrangement in the eZ80F93 . . . 194
Flash Memory Overview . . . 194
Programming Flash Memory . . . 195
Erasing Flash Memory . . . 197
Flash Control Registers . . . 197

eZ80®CPU Instruction Set . . . 208

Op-Code Map . . . 213

On-Chip Oscillators . . . 219
20 MHz Primary Crystal Oscillator Operation . . . 219
32 kHz Real-Time Clock Crystal Oscillator Operation . . . 220

Electrical Characteristics . . . 222
Absolute Maximum Ratings . . . 222
DC Characteristics . . . 223
POR and VBO Electrical Characteristics . . . 224
Typical Current Consumption Under Various Operating Conditions . . . 224
AC Characteristics . . . 229
External Memory Read Timing . . . 230
External Memory Write Timing . . . 231
External I/O Read Timing . . . 233
External I/O Write Timing . . . 234
Wait State Timing for Read Operations . . . 235
Wait State Timing for Write Operations . . . 236
General Purpose I/O Port Input Sample Timing . . . 237

SPI Flags

Mode Fault

The Mode Fault flag (MODF) indicates that there may be a multimaster conflict for system control. The MODF bit is normally cleared to 0 and is only set to 1 when the master device's $\overline{SS}$ pin is pulled Low. When a mode fault is detected, the following occurs:

1. The MODF flag (SPI_SR[4]) is set to 1.
2. The SPI device is disabled by clearing the SPI_EN bit (SPI_CTL[5]) to 0.
3. The MASTER_EN bit (SPI_CTL[4]) is cleared to 0, forcing the device into SLAVE mode.
4. If the SPI interrupt is enabled by setting IRQ_EN (SPI_CTL[7]) High, an SPI interrupt is generated.

Clearing the Mode Fault flag is performed by reading the SPI Status register. The other SPI control bits (SPI_EN and MASTER_EN) must be restored to their original states by user software after the Mode Fault flag is cleared.

Write Collision

The WRITE COLLISION flag, WCOL (SPI_SR[5]), is set to 1 when an attempt is made to write to the SPI Transmit Shift register (SPI_TSR) while data transfer occurs. Clearing the WCOL bit is performed by reading SPI_SR with the WCOL bit set.

SPI Baud Rate Generator

The SPI's Baud Rate Generator creates a lower frequency clock from the high-frequency system clock. The Baud Rate Generator output is used as the clock source by the SPI.

Baud Rate Generator Functional Description

The SPI's Baud Rate Generator consists of a 16-bit downcounter, two 8-bit registers, and associated decoding logic. The Baud Rate Generator's initial value is defined by the two BRG Divisor Latch registers, {SPI_BRG_H, SPI_BRG_L}. At the rising edge of each system clock, the BRG decrements until it reaches the value `0001h`. On the next system clock rising edge, the BRG reloads the initial value from {SPI_BRG_H, SPI_BRG_L) and outputs a pulse to indicate the end-of-count. Calculate the SPI Data Rate with the following equation:

$$\text{SPI Data Rate (bps)} = \frac{\text{System Clock Frequency}}{\text{2 X SPI Baud Rate Generator Divisor}}$$

Bit Position	Value	Description
7 SPIF	0	SPI data transfer is not finished.
	1	SPI data transfer is finished. If enabled, an interrupt is generated. This bit flag is cleared to 0 by a Read of the SPI_SR register.
6 WCOL	0	An SPI write collision is not detected.
	1	An SPI write collision is detected. This bit flag is cleared to 0 by a Read of the SPI_SR registers.
5	0	Reserved.
4 MODF	0	A mode fault (multimaster conflict) is not detected.
	1	A mode fault (multimaster conflict) is detected. This bit flag is cleared to 0 by a Read of the SPI_SR register.
[3:0]	0000	Reserved.

SPI Transmit Shift Register

The SPI Transmit Shift register (SPI_TSR) is used by the SPI master to transmit data onto the SPI serial bus to the slave device. A Write to the SPI_TSR register places data directly into the shift register for transmission. A Write to this register within an SPI device configured as a master initiates transmission of the byte of the data loaded into the register. At the completion of transmitting a byte of data, the SPIF status bit (SPI_SR[7]) is set to 1 in both the master and slave devices.

The SPI Transmit Shift Write Only register shares the same address space as the SPI Receive Buffer Read Only register. See Table 76.

Table 76. SPI Transmit Shift Register(SPI_TSR = 00BCh)

Bit	7	6	5	4	3	2	1	0
Reset	X	X	X	X	X	X	X	X
CPU Access	W	W	W	W	W	W	W	W
Note: W = Write only.								

Bit Position	Value	Description
[7:0] TX_DATA	00h–FFh	SPI transmit data.

SPI Receive Buffer Register

The SPI Receive Buffer register (SPI_RBR) is used by the SPI slave to receive data from the serial bus. The SPIF bit must be cleared prior to a second transfer of data from the shift register or an overrun condition exists. In cases of overrun the byte that caused the overrun is lost.

The SPI Receive Buffer Read Only register shares the same address space as the SPI Transmit Shift Write Only register. See Table 77.

Table 77. SPI Receive Buffer Register(SPI_RBR = 00BCh)

Bit	7	6	5	4	3	2	1	0
Reset	X	X	X	X	X	X	X	X
CPU Access	R	R	R	R	R	R	R	R
Note: R = Read Only.								

Data Validity

The data on the SDA line must be stable during the High period of the clock. The High or Low state of the data line can only change when the clock signal on the SCL line is Low as displayed in Figure 32.

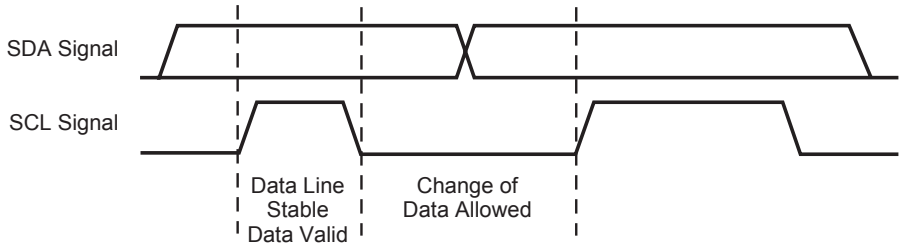

Figure 32. I²C Clock and Data Relationship

START and STOP Conditions

Within the I²C bus protocol, unique situations arise which are defined as START and STOP conditions. See Figure 33. A High-to-Low transition on the SDA line while SCL is High indicates a START condition. A Low-to-High transition on the SDA line while SCL is High defines a STOP condition.

START and STOP conditions are always generated by the master. The bus is considered to be busy after the START condition. The bus is considered to be free a defined time after the STOP condition.

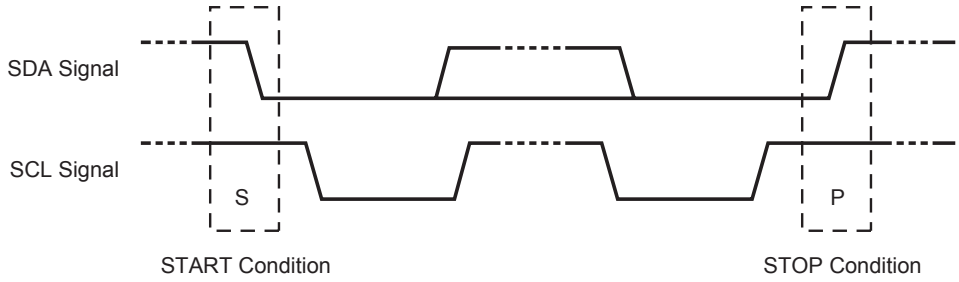

Figure 33. START and STOP Conditions In I²C Protocol

Table 85. I²C Extended Slave Address Registers(I2C_XSAR = 00C9h)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Note: R/W = Read/Write.								

Bit Position	Value	Description
[7:0] SLAX	00h–FFh	Least-significant 8 bits of the 10-bit extended slave address.

I²C Data Register

This register contains the data byte/slave address to be transmitted or the data byte just received. In transmit mode, the msb of the byte is transmitted first. In receive mode, the first bit received is placed in the msb of the register. After each byte is transmitted, the I2C_DR register contains the byte that is present on the bus in case a lost arbitration event occurs. See Table 86.

Table 86. I²C Data Registers(I2C_DR = 00CAh)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Note: R/W = Read/Write.								

Bit Position	Value	Description
[7:0] DATA	00h–FFh	I²C data byte.

I²C Control Register

The I2C_CTL register is a control register that is used to control the interrupts and the master slave relationships on the I²C bus.

When the Interrupt Enable bit (IEN) is set to 1, the interrupt line goes High when the IFLG is set to 1. When IEN is cleared to 0, the interrupt line always remains Low.

When the Bus Enable bit (ENAB) is set to 0, the I²C bus inputs SCLx and SDAx are ignored and the I²C module does not respond to any address on the bus. When ENAB is

set to 1, the I^2C responds to calls to its slave address and to the general call address if the GCE bit (I2C_SAR[0]) is set to 1.

When the Master Mode Start bit (STA) is set to 1, the I^2C enters MASTER mode and sends a START condition on the bus when the bus is free. If the STA bit is set to 1 when the I^2C module is already in MASTER mode and one or more bytes are transmitted, then a repeated START condition is sent. If the STA bit is set to 1 when the I^2C block is accessed in SLAVE mode, the I^2C completes the data transfer in SLAVE mode and then enters MASTER mode when the bus is released. The STA bit is automatically cleared after a START condition is set. Writing a 0 to this bit produces no effect.

If the Master Mode Stop bit (STP) is set to 1 in MASTER mode, a STOP condition is transmitted on the I^2C bus. If the STP bit is set to 1 in slave move, the I^2C module operates as if a STOP condition is received, but no STOP condition is transmitted. If both STA and STP bits are set, the I^2C block first transmits the STOP condition (if in MASTER mode) and then transmit the START condition. The STP bit is cleared automatically. Writing a 0 to this bit produces no effect.

The I^2C Interrupt Flag (IFLG) is set to 1 automatically when any of 30 of the possible 31 I^2C states is entered. The only state that does not set the IFLG bit is state F8h. If IFLG is set to 1 and the IEN bit is also set, an interrupt is generated. When IFLG is set by the I^2C, the Low period of the I^2C bus clock line is stretched and the data transfer is suspended. When a 0 is written to IFLG, the interrupt is cleared and the I^2C clock line is released.

When the I^2C Acknowledge bit (AAK) is set to 1, an Acknowledge is sent during the acknowledge clock pulse on the I^2C bus if:

- Either the whole of a 7-bit slave address or the first or second byte of a 10-bit slave address is received
- The general call address is received and the General Call Enable bit in I2C_SAR is set to 1
- A data byte is received while in MASTER or SLAVE modes

When AAK is cleared to 0, a NACK is sent when a data byte is received in MASTER or SLAVE mode. If AAK is cleared to 0 in the Slave Transmitter mode, the byte in the I2C_DR register is assumed to be the final byte. After this byte is transmitted, the I^2C block enter states `C8h`, then returns to the idle state. The I^2C module does not respond to its slave address unless AAK is set. See Table 87 on page 157.

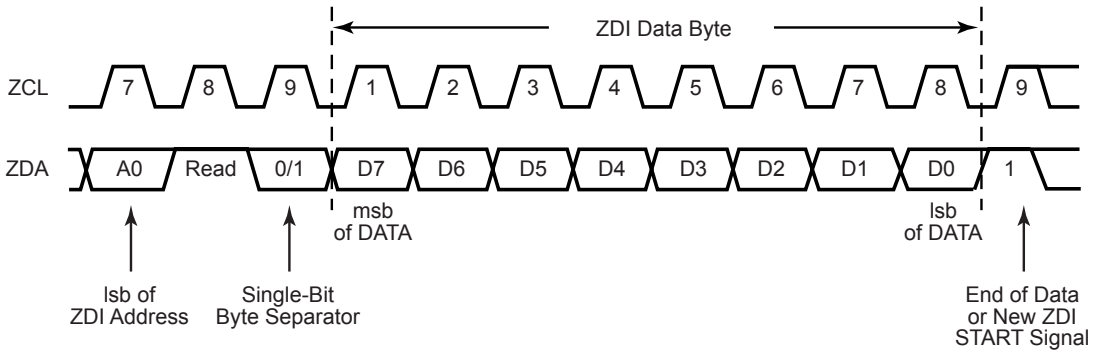

Figure 44.ZDI Single-Byte Data Read Timing

ZDI Block Read

A Block Read operation is initiated the same as a single-byte Read; however, the ZDI master continues to clock in the next byte from the ZDI slave as the ZDI slave continues to output data. The ZDI register address counter increments with each Read. If the ZDI register address reaches the end of the Read Only ZDI register address space (`20h`), the address stops incrementing. Figure 45 displays the ZDI's Block Read timing.

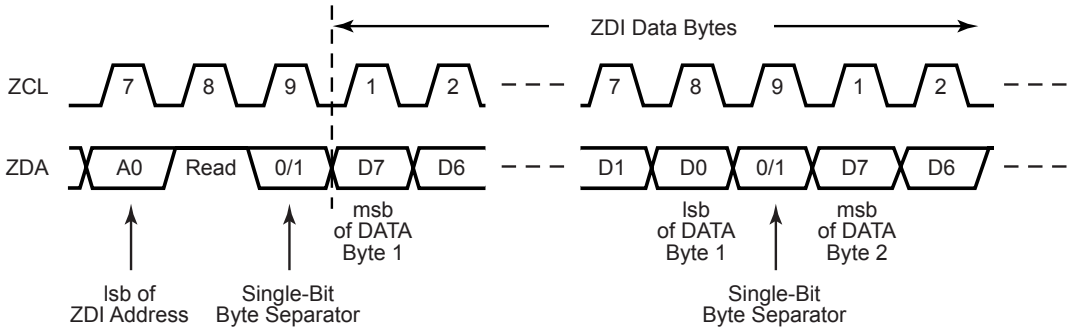

Figure 45.ZDI Block Data Read Timing

Operation of the eZ80F92 Device During ZDI Breakpoints

If the ZDI forces the CPU to BREAK, only the CPU suspends operation. The system clock continues to operate and drive other peripherals. Those peripherals that can operate autonomously from the CPU may continue to operate, if so enabled. For example, the Watchdog Timer and Programmable Reload Timers continue to count during a ZDI BREAK point.

When using the ZDI interface, any Write or Read operations of peripheral registers in the I/O address space produces the same effect as Read or Write operations using the CPU.

ZDI Bus Status Register

The ZDI Bus Status register monitors BUSACKs during ZDI DEBUG mode. See Table 108.

Table 108. ZDI Bus Control Register(ZDI_BUS_STAT = 17h in the ZDI Register Read Only Address Space)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R	R	R	R	R	R	R	R
Note: R = Read Only.								

Bit Position	Value	Description
7 ZDI_BUSACK_EN	0	Bus requests by external peripherals using the BUSREQ pin are ignored. The bus acknowledge signal, BUSACK, is not asserted.
	1	Bus requests by external peripherals using the BUSREQ pin are accepted. A bus acknowledge occurs at the end of the current ZDI operation. The bus acknowledge is indicated by asserting the BUSACK pin.
6 ZDI_BUS_STAT	0	Address and data buses are not relinquished to an external peripheral. bus acknowledge is deasserted (BUSACK pin is High).
	1	Address and data buses are relinquished to an external peripheral. bus acknowledge is asserted (BUSACK pin is Low).
[5:0]	000000	Reserved.

ZDI Read Memory Register

When a Read is executed from the ZDI Read Memory register, the eZ80F92 device fetches the data from the memory address currently pointed to by the program counter, PC; the program counter is then incremented. In Z80® MEMORY mode, the memory address is {MBASE, PC[15:0]}. In ADL MEMORY mode, the memory address is PC[23:0]. Refer to the *eZ80® CPU User Manual (UM0077)* for more information regarding Z80® and ADL MEMORY modes. The program counter, PC, increments after each data Read. However, the ZDI register address does not increment automatically when this register is accessed. As a result, the ZDI master can read any number of data bytes out of memory through the ZDI Read Memory register. See Table 109 on page 186.

On-Chip Instrumentation

Introduction to On-Chip Instrumentation

On-Chip Instrumentation[1] (OCI™) for the eZ80® CPU core enables powerful debugging features. The OCI provides run control, memory and register visibility, complex break-points, and trace history features.

The OCI employs all of the functions of the ZDI as described in the Zilog Debug Interface section that starts on page 162. It also adds the following debug features:

- Control via a 4-pin Joint Test Action Group (JTAG)-standard port that conforms to the IEEE Standard 1149.1 (Test Access Port and Boundary Scan Architecture)[2]
- Complex break point trigger functions
- Break point enhancements, such as the ability to:
  - Define two break point addresses that form a range
  - Break on masked data values
  - Start or stop trace
  - Assert a trigger output signal
- Trace history buffer
- Software break point instruction

There are four sections to the OCI:

1. JTAG interface
2. ZDI debug control
3. Trace buffer memory
4. Complex triggers

OCI Activation

OCI features clock initialization circuitry so that external debug hardware can be detected during power-up. The external debugger must drive the OCI clock pin (TCK) Low at least two system clock cycles prior to the end of the RESET to activate the OCI block. If TCK is High at the end of the RESET, the OCI block shuts down so that it does not draw power in normal product operation. When the OCI is shut down, ZDI is enabled directly and can

1. On-Chip Instrumentation and OCI are trademarks of First Silicon Solutions, Inc.
2. The eZ80F92 does not contain the boundary scan register required for 1149.1 compliance.

Table 121. Flash Page Select Register; (FLASH_PAGE=00FCh)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Note: R/W = Read/Write, R = Read Only.								

Bit Position	Value	Description
[7] INFO_EN	0	Flash accesses main Flash memory.
	1	Flash accesses the Information Page. Page Erase and Mass Erase operations affect the Information Page only.
[6:0]* FLASH_PAGE	00h–7Fh	Page address of Flash memory to be used during the Page Erase or I/O Write of the main Flash memory. When INFO_EN is set to 1, this field is ignored.

Note: *Only 6 bits are available in the eZ80F93 device.

Flash Row Select Register

The Flash Row Select register is a 3-bit value used to define one of the eight rows of Flash memory on a single page. This register is used for all I/O Write access to Flash.

Table 122. Flash Row Select Register; (FLASH_ROW=00FDh)

Bit	7	6	5	4	3	2	1	0
Reset	X	X	X	X	X	0	0	0
CPU Access	R	R	R	R	R	R/W	R/W	R/W
Note: R/W = Read/Write, R = Read Only.								

Bit Position	Value	Description
[7:3]	00h	Reserved.
[2:0] FLASH_ROW	0h–7h	Row address of Flash memory to be used during an I/O Write to Flash memory. When INFO_EN is 1 in the Flash Page Select Register, values for this field are restricted to 0h–1h, which selects between the two rows in the Information Page.

Table 149. External Write Timing

Parameter	Abbreviation	Delay (ns) Min	Delay (ns) Max
T_1	Clock Rise to ADDR Valid Delay	—	13
T_2	Clock Rise to ADDR Hold Time	2.0	—
T_3	Clock Fall to Output DATA Valid Delay	—	11
T_4	Clock Rise to DATA Hold Time	2.0	—
T_5	Clock Rise to $\overline{CSx}$ Assertion Delay	2.0	19.0
T_6	Clock Rise to $\overline{CSx}$ Deassertion Delay	2.0	18.0
T_7	Clock Rise to $\overline{MREQ}$ Assertion Delay	2.0	16.0
T_8	Clock Rise to $\overline{MREQ}$ Deassertion Delay	2.0	16.0
T_9	Clock Fall to $\overline{WR}$ Assertion Delay	1.8	6.5
T_{10}	Clock Rise to $\overline{WR}$ Deassertion Delay*	1.6	6.5
	$\overline{WR}$ Deassertion to ADDR Hold Time	0.25	—
	$\overline{WR}$ Deassertion to DATA Hold Time	0.25	—
	$\overline{WR}$ Deassertion to $\overline{CSx}$ Hold Time	0.25	—
	$\overline{WR}$ Deassertion to $\overline{MREQ}$ Hold Time	0.25	—

Note: *At the conclusion of a Write cycle, deassertion of $\overline{WR}$ always occurs before any change to ADDR, DATA, $\overline{CSx}$, or $\overline{MREQ}$.

General Purpose I/O Port Input Sample Timing

Figure 64 displays timing of the GPIO input sampling. The input value on a GPIO port pin is sampled on the rising edge of the system clock. The port value is then available to the CPU on the second rising clock edge following the change of the port value.

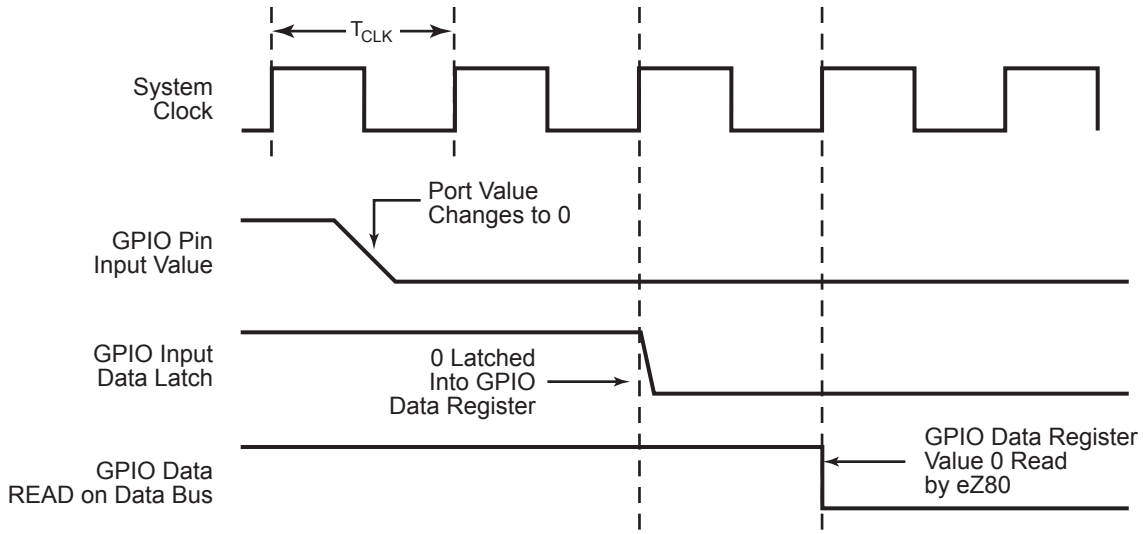

Figure 64.Port Input Sample Timing

Table 152. GPIO Port Output Timing

		Delay (ns)	
Parameter	**Abbreviation**	**Min**	**Max**
T_1	Clock Rise to Port Output Delay	2.0	15.0

Interrupt Enable Flag 184
Interrupt Enable flags 47
interrupt input 13, 14, 15, 16, 18, 19
interrupt request 41, 42, 43, 45, 46, 47, 82, 203
interrupt service routine 45, 47
interrupt service routine, SPI 45
interrupt vector 45, 46
interrupt vector address 46, 47
interrupt, highest-priority 45, 46
interrupts, edge-selectable 43
Introduction to On-Chip Instrumentation 187
Introduction, Zilog Debug Interface 162
IORQ 11, 12, 22, 51, 53, 54, 56, 57, 60
IORQ Assertion Delay 233, 234
IORQ Deassertion Delay 233, 234
IORQ Hold Time 235
IR_RxD 13, 126, 127, 128
IR_RxD modulation signal 13, 125, 128, 129
IR_TxD 13
IR_TxD modulation signal 13, 125, 128, 129
IrDA Encoder/Decoder 128
IrDA encoder/decoder 13
IrDA endec 37
IrDA specifications 124
IrDA standard 124
IrDA standard baud rates 124
IrDA transceiver 128
IrDA Transmit Data 13
IrDA—see Infrared Data Association 124
IRQ 46
irq_en 82, 134, 137
irq_en bit 79
IVECT 45, 46, 47

J

Jitter, Infrared Encoder/Decoder 128
JTAG interface 187
JTAG mode selection 188
JTAG Test Clock 12
JTAG Test Data In 12
JTAG Test Data Out 12
JTAG Test Mode 12
JTAG Test Trigger Output 12

L

least-significant bit 105, 164
least-significant byte 46, 84
level-sensitive interrupt input 128
level-sensitive interrupt modes 41
level-sensitive interrupts 43
Level-Triggered Interrupts 42
Line break detection 104
Loopback Testing, Infrared Encoder/Decoder 128
low-byte vector 45
Low-Power Modes 35
LSB 160
lsb 84
lsb—see least-significant bit 83, 84, 146, 147, 149, 152
LSB—see least-significant byte 46, 47, 84

M

maskable interrupt 36
maskable interrupt sources 45
maskable interrupt vectors 46
Maskable Interrupts 45
Mass Erase 197
Mass Erase operation 202, 203, 205, 206, 207
Mass Erase Violation 203
Master In, Slave Out 19, 131
MASTER mode 132, 141, 156, 158, 159, 160, 161
Master mode 152, 157
master mode 151
Master Mode Start bit 156
Master Mode Stop bit 156
MASTER mode, SPI 133
Master Out, Slave In 19, 131
Master Receive 141, 149
Master Transmit 146
MASTER TRANSMIT mode 141
master_en bit 134
Memory and I/O Chip Selects 48
Memory Chip Select Example 49
Memory Chip Select Operation 48
Memory Chip Select Priority 49
Memory Request 11
memory space 48, 50

Programming Flash Memory 195
pull-up resistor, external 40, 141
PUSH, Op Code Map 213, 215, 217

R

RAM 190
RAM Address Upper Byte Register 192
RAM Control Register 192
Random Access Memory 190
RD 11, 22, 48, 51, 53, 56, 57, 60
RD Assertion Delay 231, 233
RD Deassertion Delay 231, 233
Reading the Current Count Value 79
Real-Time Clock 32, 35, 88, 219, 220
Real-Time Clock Alarm 89
Real-Time Clock alarm 35
Real-Time Clock Alarm Control Register 102
Real-Time Clock Alarm Day-of-the-Week Register 101
Real-Time Clock Alarm Hours Register 100
Real-Time Clock Alarm Minutes Register 99
Real-Time Clock Alarm Seconds Register 98
Real-Time Clock Battery Backup 89
Real-Time Clock Century Register 97
Real-Time Clock circuit 224
Real-Time Clock Control Register 102
Real-Time Clock Crystal Input 12
Real-Time Clock Crystal Output 12
Real-Time Clock Day-of-the-Month Register 94
Real-Time Clock Day-of-the-Week Register 93
Real-Time Clock Hours Register 92
Real-Time Clock Minutes Register 91
Real-Time Clock Month Register 95
Real-Time Clock Oscillator and Source Selection 89
Real-Time Clock Overview 88
Real-Time Clock Power Supply 12
Real-Time Clock Recommended Operation 89
Real-Time Clock Registers 90
Real-Time Clock Seconds Register 90
Real-Time Clock source 72, 74, 76, 80, 86, 87
Real-Time Clock Year Register 96
Receive, Infrared Encoder/Decoder 125
Recommended Usage of the Baud Rate Generator 110
Register Map 25
Request to Send 13, 16, 119
RESET 11, 22, 32, 33, 35, 36, 40, 48, 72, 73, 74, 89, 90, 102, 109, 110, 111, 129, 135, 173, 175, 187, 188, 190, 192, 200, 202, 224
Reset 32
Reset controller 32, 33
RESET event 39
RESET mode timer 32
Reset Operation 32
RESET Or NMI Generation 73
Reset States 49
Resetting the I2C Registers 153
RI 106, 119, 122
RI0 15, 128
RI1 17, 41
Ring Indicator 15, 17, 122
Row Program Time-out 203
rst_flag bit 73
RTC 12, 29, 30, 76, 88
RTC alarm condition 102
RTC alarm registers 90
RTC clock source 103
RTC count 89, 102
RTC count registers 90
RTC Supply Current 224
RTC Supply Voltage 223
RTC_UNLOCK 103
RTC_UNLOCK bit 89, 90, 102
RTC_VDD 12, 22
RTC_XIN 12, 22
RTC_XOUT 12, 22
RTS 119, 122, 128
RTS0 13
RTS1 16
RxD0 13
RxD1 15

S

Schmitt Trigger 11
Schmitt Trigger Input 20